

AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

1. (Currently Amended) A method, comprising:

retrieving a set of ~~programming statements~~ allocate buffer instructions from a storage device associated with a multithreaded network processing element, the network processing element having a local memory;

arranging for a first portion of the local memory to be allocated to a first thread context in accordance with a ~~programming statement~~ an allocate buffer instruction that is associated with a first thread and symbolically references a buffer name, the ~~programming statement~~ allocate buffer instruction includes an indication of a read/write status of the first portion; and

arranging for a second portion of the local memory to be allocated to a second thread context in accordance with a ~~programming statement~~ an allocate buffer instruction that is associated with a second thread and symbolically references the buffer name, the ~~programming statement~~ allocate buffer instruction includes an indication of a read/write status of the second portion; and

wherein the symbolically referenced buffer name includes both letters and numbers.

2. (Currently Amended) The method of claim 1, further comprising:

including a base, size, and offset with the ~~programming statement~~ allocate buffer instruction associated with the first thread.

3. (Currently Amended) The method of claim 1, wherein said arranging comprises translating the ~~programming statements~~ allocate buffer instructions into code.

4. (Original) The method of claim 1, further comprising:
arranging for information associated with the first thread context to be stored in the first portion of the local memory; and
arranging for information associated with the second thread context to be stored in the second portion of the local memory.
5. (Currently Amended) The method of claim 1, further comprising:
freeing the second portion of the local memory in accordance with another programming statement allocate buffer instruction that symbolically references the buffer name.
6. (Original) The method of claim 1, wherein the symbolic reference to the buffer name may be passed in at least one of: (i) a function, and (ii) a macro.
7. (Currently Amended) The method of claim 1, further comprising:
translating the set of programming statements allocate buffer instructions into code; and
providing the code.
8. (Original) The method of claim 7, wherein the provided code is associated with at least one of: (i) assembly language, and (ii) microcode.
9. (Original) The method of claim 1, wherein the local memory comprises at least one of: (i) memory at the network processing element, (ii) hardware registers at the network processing element, and (iii) a local cache.

10. (Original) The method of claim 1, wherein the network processing element is a reduced instruction set computer microengine in a network device.

11. (Original) The method of claim 10, wherein the network device is associated with at least one of: (i) information packet header parsing, (ii) exception packet identification, (iii) information packet receipt, (iv) information packet transformation, and (v) information packet transmission.

12. (Original) The method of claim 10, wherein the network device is associated with at least one of: (i) Internet protocol information packets, (ii) Ethernet information packets, (iii) asynchronous transfer mode protocol, (iv) a local area network, (v) a wide area network, (vi) a network processor, (vii) a switch, and (viii) a router.

13. (Cancelled)

14. (Currently Amended) An article, comprising:
a storage medium having stored thereon instructions that when executed by a machine result in the following:

retrieving a set of ~~programming statements~~ allocate buffer instructions from a storage device associated with a multithreaded network processing element, the network processing element having a local memory;

arranging for a first portion of the local memory to be allocated to a first thread context in accordance with a ~~programming statement~~ an allocate buffer instruction that is associated with a first thread and symbolically references a buffer name, ~~the programming statement~~ allocate buffer instruction includes an indication of a read/write status of the first portion; and

arranging for a second portion of the local memory to be allocated to a second thread context in accordance with a ~~programming statement~~ an allocate buffer instruction that is associated with a second thread and symbolically references the buffer name, the ~~programming statement~~ allocate buffer instruction includes an indication of a read/write status of the second portion; and

wherein the symbolically referenced buffer name includes both letters and numbers.

15. (Currently Amended) The article of claim 14, wherein execution of the instructions further results in:

translating the set of ~~programming statements~~ allocate buffer instructions into code.

16. (Original) The article of claim 15, wherein execution of the instructions further results in:

providing the code.

17. (Currently Amended) An article, comprising:

a processor; and

a storage medium having stored thereon ~~storing~~ a set of ~~programming statements~~ allocate buffer instructions ~~adapted to be translated into code~~, said ~~programming statements~~ allocate buffer instructions including:

~~a programming statement~~ an allocate buffer instruction associated with a first thread that includes an indication of a read/write status and uses a buffer name to symbolically reference information that a first thread context will store in local memory at a multithreaded network processing element; and

~~a programming statement~~ an allocate buffer instruction associated with a second thread that includes an indication of a read/write status and uses the buffer name to

symbolically reference information that a second thread context will store in the local memory at the network processing element,

wherein the symbolically referenced buffer name includes both letters and numbers.

18. (Cancelled)

19. (Currently Amended) A method, comprising:

defining a ~~programming statement~~ an allocate buffer instruction ~~that is associated with a~~ first thread that includes an indication of a read/write status and uses a buffer name to symbolically reference information that a first thread context will store in local memory at a multithreaded network processing element; and

defining a ~~programming statement~~ an allocate buffer instruction ~~that is associated with a~~ second thread that includes an indication of a read/write status and uses the buffer name to symbolically reference information that a second thread context will store in the local memory at the network processing element;

wherein the symbolically referenced buffer name includes both letters and numbers.

20. (~~Original~~ Currently Amended) The method of claim 19, further comprising:
arranging for the ~~programming statements~~ allocate buffer instruction to be translated into code.

21. (Currently Amended) A system, comprising:

a processor; and

a hard disk drive having stored therein instructions that when executed by a machine result in the following:

translating C programming language instructions into code, and
translating an additional ~~programming statement~~allocate buffer instruction into
code, the additional ~~programming statement~~allocate buffer instruction including an
indication of a read/write status and using a buffer name to symbolically reference
information that a thread context will store in local memory at a multithreaded network
processing element,

wherein the symbolically referenced buffer name includes both letters and
numbers.

22. (Original) The system of claim 21, further comprising:
an interface to facilitate a transfer of the code from the system to the network processing element.